

09/29/00
jc535 U.S. PTO
REGISTERED PATENT ATTORNEY (33,619)

CHRISTOPHER H. LYNT
Attorney at Law
Patent, Trademark and Copyright Law

7502 Toll Court
Alexandria, VA 22306
United States of America

TELEPHONE: (703) 768-0956
TELEFAX: (703) 721-3327
E-MAIL: chlpatent@aol.com

jc511 U.S. PTO
09/672043
09/29/00

September 29, 2000

Box PATENT APPLICATION

Assistant Commissioner for Patents
Washington, D.C. 20231

Attorney Reference: ROC920000177-US1 (IBM-150)

Re: New Patent Application of: SCHEEL et al.

Title: **TECHNIQUE FOR CONFIGURING PROCESSORS IN SYSTEM WITH LOGICAL PARTITIONS**

Sir:

Please find attached hereto an application for patent which includes:
Specification, Claims, Declaration, and Power of Attorney.
Drawings: 2 sheets of drawings (Figs. 1-2)
Fee (see formula below)

Basic Fee \$345/690 \$ 690.00

Additional Fees:

Total number of claims: 14
in excess of 20: 0 times \$9/18 \$.00

Number of independent claims: 3
in excess of 3: times \$39/78 \$.00

Multiple Dependent Claims \$130/260 \$ 0.00

An Assignment and recordation cover sheet are likewise enclosed;

Recording Fee \$40. \$ 40.00

TOTAL FEES FOR THE ABOVE APPLICATION \$ 730.00

Please charge the above fees to IBM Corporation Deposit Account No. 09-0465 and notify us accordingly.

Respectfully submitted,



Christopher H. Lynt (Registration No. 33,619)

CHL:chl

09672043.092900

TECHNIQUE FOR CONFIGURING PROCESSORS IN SYSTEM WITH LOGICAL PARTITIONS

Background Of The Invention

5 Field of The Invention

The invention relates to the field of multiprocessing, and in particular, to the distribution of resources on a system where multiple operating system images (same or different) are involved.

10 Background Information

006260" CH024960
In early computer systems, a single set of program instructions was used to control the basic computer system operations such as input/output, disk access, etc. This set of program instructions was referred to as the 'operating system.' Early computers also generally had only a single central processing unit or CPU where the program instructions were executed. Modern computer systems often have multiple processing units and/or are capable of running multiple operating systems at the same time. Each of the multiple operating systems running on such a modern computer may be confined to operate in a respective logical partition, i.e., an processing space reserved for that operating system. In a shared system, one user may be running the Windows operating system, while another user may be running Unix, for example. Also, there could be multiple users running the same operating system, e.g., Unix, but each running their own copy or 'image' in different respective partitions of the system. These partitions serve to prevent the operating systems running concurrently from interfering with each other.

On a modern computing system running multiple operating system images (i.e., the same operating system or different operating systems) as logical systems in respective partitions, one of the key items to be distributed between each logical system is processing resources. These resources can be dedicated to a particular partition or shared among

multiple partitions. Configuration of these resources takes many user interface forms with GUI (graphical user interface) being the most common.

Even the earliest computer system included a rudimentary mechanism that allowed it to receive input from its users. These mechanism are often called user interfaces. The user interfaces of early computers often employed lights and switches that allowed the computer system user to communicate with the computer system in the system's native language. While this form of communication was effective, its use was essentially limited to scientists and engineers who were trained to understand the particulars of the computer system. For this reason, computer system providers moved to user interfaces that provided for entry of textual commands. While these command-based user interfaces were easier to use than their predecessors, they still typically involved a large number of commands that were often difficult for non-technical individuals to understand and remember. The shortcomings of command-based user interfaces led to the introduction of graphical user interfaces, which are often referred to as GUIs. Instead of requiring the user to have in-depth knowledge of the computer system, graphical user interfaces allow the every day user to communicate with the computer system via a pointing device such as a mouse. To use the computer system to perform various tasks, the user simply "points and clicks" on various GUI items such as icons and taskbars. These icons and taskbars are user friendly ways of activating different programs on the computer.

Techniques for managing the allocation of the processing resources within the user interface generally requires allocating a quantity of the processors. For dedicated processors, the quantity is generally a whole number representing the number of processors. For shared processors, the quantity can be either a numerical value representing both whole and partial processing units (i.e., 2.33 to describe two and a third processors) or a weighting scheme indicating a percentage of the shared

processors (i.e., a partition weight of 100 with all partition weights totaling 1000 for 5 shared processors to indicated a half of a processor). These techniques are user interface independent values which could be applied to a graphical interface, a command line interface, or a menu driven text interface.

The problem with these techniques is that they do not reflect the real life situation of how users think of the processors in their system. System administrators and operators generally think of the processing capacity of their system in a workload measurement value not the number of processors.

There are many known measurements of the processing capacity of the system. One simple measure is MIPS or million instructions per second. Other more sophisticated measures are derived from industry standard workloads such as TPC-C (a specific workload defined by the Transaction Processing Performance Council) and CPW (commercial processing workload). These workloads values are generally well known for server and mainframe class systems and are commonly used to describe system performance in marketing and sales literature.

Configuring logical partitions using CPW values is a technique employed in the V5R1 logical partitioning GUI (graphical user interface). This GUI (a component of the IBM™ AS/400™ Client Access Express for Windows licensed program product), however, will not make suggestions of dedicated versus shared processors.

Therefore, a way to overcome the above problems which would enable system administrators to configure processors for logical partitions using a capacity-based measurement, and which would suggest whether to use dedicated or shared processors, would make the computer systems more user friendly and the system administrators more productive.

Summary Of The Invention

It is, therefore, a principal object of this invention to provide a method and apparatus for configuring processors in a system with logical partitions

5 It is another object of the invention to provide a method and apparatus that solves the above mentioned problems so that configuration of logical partition processors is simplified.

These and other objects of the present invention are accomplished by the method and apparatus disclosed herein.

10 According to an aspect of the invention, a capacity-based measurement technique is used to configure processors for logical partitions.

According to an aspect of the invention, instead of specifying a quantity of processing units for a logical partition, the user interface
15 accepts a processing capacity value in units of work, such as CPW or TPC-C, or any well-defined workload measurement units.

According to an aspect of the invention, an exemplary routine for implementing the invention includes allowing a user to select the units to use for configuring the processors, e.g., CPW or TPC-C, or any well-
20 defined workload measurement units.

According to an aspect of the invention, the exemplary routine also prompts the user for the quantity of processing power required, i.e., the partition workload capacity required. These two steps can be combined into one step depending on the design of the user interface. That is, the
25 user can be prompted to fill-in the numerical quantity and units on the same screen, or to fill-in the numerical value next to the appropriate units to be used, for example.

According to an aspect of the invention, using a look-up table, for example, the system work capacity for the target system is obtained in the
30 appropriate units from the preceding steps.

According to an aspect of the invention, the number of partition processors is then calculated using the formula the number of partition processors equals the total number of system processors, times the partition workload capacity divided by the system work capacity.

5 According to an aspect of the invention, the resolution for the number of partition processors result is preferably two digits to the right of the decimal, i.e., two decimal places.

10 According to an aspect of the invention, the calculated number of partition processors is then tested to see if it is within a predetermined percentage, e.g., twenty-five percent, of the next full processor increment. If within the predetermined percentage, then the routine recommends using dedicated processors. Otherwise, using shared processors is recommended.

15 According to an aspect of the invention, the calculated number of partition processors and the recommended mode, i.e., shared versus dedicated processors, are displayed to the user so that the user can validate or change the values.

20 According to an aspect of the invention, after validation, the processors are configured according to the settings determined by the routine.

25 According to an aspect of the invention, a system administrator enters the partition's processing capacity, and the user interface and logical partition management code performs any necessary calculations to determine the number of processing units.

30 According to an aspect of the invention, dedicated or shared processing resources may be recommended based on predetermined criteria.

According to an aspect of the invention, the technique is independent of the type of user interface.

30 According to an aspect of the invention, the configuration of logical partition processors is simplified by utilizing output from the manual

system planning exercise (a workload capacity value) into configuration for the partition directly without any intermediate calculations.

These and other aspects of the invention will become apparent from the detailed description set forth below.

5

Brief Description Of The Drawings

Fig. 1 illustrates a flow chart of a process according to an exemplary embodiment of the present invention.

Fig. 2 is a block diagram of a computer system for implementing a technique for configuring processors according to an exemplary embodiment of the invention.

10

Detailed Description Of The Preferred Embodiment(s)

The invention will now be described in more detail by way of example with reference to the embodiment(s) shown in the accompanying figures. It should be kept in mind that the following described embodiment(s) is/are only presented by way of example and should not be construed as limiting the inventive concept to any particular physical configuration.

15

This invention relates to a technique of using a capacity-based measurement to configure processors for logical partitions. Instead of specifying a quantity of processing units for a logical partition, the user interface accepts a processing capacity value in units of work like CPW or TPC-C, or any well-defined workload measurement units.

20

For example, a server system may have 2 processors and be capable of 1000 CPW. The system administrator wants to create a logical partition to support a 250 CPW workload.

25

To implement this partition using existing user interface techniques, the system administrator would need to utilize the following formula:

30

number of partition processors = (total number of system
processors) * (partition workload capacity) / (system work capacity)

5 This would indicate that 0.5 processing units or half a processor
are required.

To implement this partition using the invention technique, the
system administrator would enter the partitions processing capacity in
CPW as 250. The user interface and logical partition management code
would then perform any necessary calculations to determine the number
10 of processing units and could even suggest dedicated or shared
processing resources.

This technique is independent of the type of user interface.

An exemplary routine for implementing the invention could take the
following form, as illustrated in Fig. 1. The routine initially allows the user
15 to select the units to use for configuring the processors, e.g., CPW or
TPC-C, or any well-defined workload measurement units. The routine
also prompts the user for the quantity of processing power required, i.e.,
the partition workload capacity required. These two steps could be
combined into one step depending on the design of the user interface.
20 That is, the user could be prompted to fill-in the numerical quantity and
units on the same screen, or to fill-in the numerical value next to the
appropriate units to be used, for example.

The user selections are limited by what types of units are used in
the look-up table. If a new workload standard were to be used, the user
25 interface could be adapted and the table changed.

Using a look-up table, for example, the system work capacity for
the target system is obtained in the appropriate units from the preceding
steps. The context indicates the model of the target system.

If the target system model is not stored in the database, the user
30 can enter the system work capacity manually, e.g., from sale literature.

For the next calculation, the partition workload capacity and the system workload capacity are expressed using the same units, e.g., both in TPC-C or CPW, or any well-defined workload measurement units.

5 The system work capacity of the target system is stored on the target system or the configuring system.

Systems generally come with this value pre-stored somewhere on-board, e.g., in VPD (vital Product Data) storage, or VPD can enable 'client' lookup, i.e., system model 'xyz' = 123 CPW, for example, model vs. capacity. However, one skilled in the art would understand that there are
10 many different ways in which the system capacity could be stored and accessed.

There is no conversion formula for converting from one unit of processing capacity to another. The table would store capacity values in all the different units needed.

15 The number of partition processors is then calculated using the formula previously described, i.e., the number of partition processors equals the total number of system processors, times the partition workload capacity divided by the system work capacity. The resolution for the result would preferably be two digits to the right of the decimal.

20 Then the calculated number of partition processors is tested to see if it is within a predetermined percentage, e.g., twenty-five percent, of the next full processor increment. If within the predetermined percentage, then the routine will recommend using dedicated processors. For example if the result were 0.80 partition processors then dedicated
25 processors would be recommended. Otherwise, using shared processors would be recommended. The factors that determine the percentage to use as a test for shared processors are based on a definition of what is considered 'reasonable.' A good user interface allows selecting this value under an "Advanced" parameter.

30 The calculated number of partition processors and the recommended mode, i.e., shared versus dedicated processors, is

displayed to the user so that the user can validate or change the values. After validation, the processors are configured according to the settings determined by the routine. This invention simplifies the configuration of logical partition processors.

5 Fig. 2 is a block diagram of a basic computer system 200 for implementing a technique for configuring processors according to an exemplary embodiment of the invention. The basic computer system 200 includes partition processing means 201, storage means 202 in which a program implementing a graphical user interface for configuring
10 processors 203 as described above is disposed, an input device 204, e.g., mouse and/or keyboard, and a display means 205.

 The invention may be embodied as a computer program product which includes a recording medium, such as a floppy disk, a high-capacity read only memory in the form of an optically read compact disc or CD-
15 ROM, a tape, a transmission type media, such as a digital or analog communications link, or a similar computer program product.

 It will be apparent to one skilled in the art the manner of making and using the claimed invention has been adequately disclosed in the above-written description of preferred embodiments taken together with
20 the drawings. The above-described preferred embodiments of the invention are susceptible to modifications, changes, and adaptations, and the same are intended to be comprehended within the meaning and range of equivalents of the appended claims. Although equivalent components may have been mentioned herein which could be used in place of the
25 components illustrated and described with reference to the preferred embodiment(s), this is not meant to be an exhaustive treatment of all the possible equivalents, nor to limit the invention defined by the claims to any particular equivalent or combination thereof. A person skilled in the art would realize that there may be other equivalent components presently
30 known, or to be developed, which could be used within the spirit and scope of the invention defined by the claims.

1 What is claimed is:

2

1 1. A method of configuring processors in a target system,
2 comprising:

3 prompting a user to select workload units to use in the
4 configuration of the processor in the target system;

5 prompting the user to input a quantity of processing power required
6 in terms of partition workload capacity required;

7 obtaining a system work capacity for the target system in the
8 appropriate units from a look-up table;

9 calculating the number of partition processors;

10 wherein the number of partition processors equals the total number
11 of system processors, times the partition workload capacity divided by the
12 system work capacity;

13 testing the calculated number of partition processors to see if it is
14 within a predetermined percentage of the next full processor increment;

15 if within the predetermined percentage, then recommending using
16 dedicated processors, otherwise recommending using shared processors;

17 displaying the calculated number of partition processors and the
18 recommended use of dedicated or shared processors to the user for
19 validation or changing of the values; and

20 after validation, configuring the target system processors according
21 to the settings determined by the routine.

22

1 2. The method according to claim 1, wherein the workload units
2 to use in the configuration of the processor in the target system are in
3 commercial processing workload (CPW) units, transaction processing
4 performance council (TPC-C) units, or any well-defined workload
5 measurement units.

6

1 3. The method according to claim 1, wherein the number of
2 partition processors calculated has a resolution of at least two digits to the
3 right of the decimal.

1 4. The method according to claim 1, wherein the
2 predetermined percentage of the next full processor increment is twenty-
3 five percent.

1 5. A processing system running multiple operating system
2 images (same or different) having logical partitions and implementing the
3 method according to claim 1.

1 6. A computer program product, comprising:
2 a recording medium; and
3 instruction means, disposed on the recording medium, for causing
4 a computer to implement the method of configuring processors in a target
5 system according to claim 1.

1 7. A computer system having processing means, storage
2 means, input means, and display means, and operating a graphical user
3 interface utilizing the method according to claim 1.

1 8. A graphical user interface comprising:
2 means for prompting a user to select workload units to use in
3 configuration of processors in a target system;
4 means for prompting the user to input a quantity of processing
5 power required in terms of partition workload capacity required;
6 means for obtaining a system work capacity for the target system in
7 the appropriate units from a look-up table;
8 means for calculating the number of partition processors;

wherein the number of partition processors equals the total number of system processors, times the partition workload capacity divided by the system work capacity;

means for testing the calculated number of partition processors to see if it is within a predetermined percentage of the next full processor increment;

means for recommending using dedicated processors if within the predetermined percentage, and otherwise recommending using shared processors;

means for displaying the calculated number of partition processors and the recommended use of dedicated or shared processors to the user for validation or changing of the values; and

means for configuring the target system processors according to the settings determined by the above means after validation.

9. The graphical user interface according to claim 8, wherein the workload units to use in the configuration of the processor in the target system are in commercial processing workload (CPW) units, transaction processing performance council (TPC-C) units, or any well-defined workload measurement units.

10. The graphical user interface according to claim 8, wherein the number of partition processors calculated has a resolution of two digits to the right of the decimal.

11. The graphical user interface according to claim 8, wherein the predetermined percentage of the next full processor increment is twenty-five percent.

1 12. A processing system running multiple operating system
2 images (same or different) having logical partitions and implementing the
3 a graphical user interface according to claim 8.

4
1 13. A computer program product, comprising:
2 a recording medium; and
3 instruction means, disposed on the recording medium, for causing
4 a computer to implement the graphical user interface according to claim 8.

5
1 14. A computer system comprising:
2 partition processing means for running multiple operating system
3 images;
4 storage means for storing a program and data;
5 an input device for inputting data; and
6 display means for displaying graphical representations to a user;
7 wherein the program implements the graphical user interface for
8 configuring processors.

9

Abstract

Configuring processors in a target system includes prompting a user to select workload units to use in the configuration, prompting the user to input a quantity of processing power required in terms of partition workload capacity required, obtaining a system work capacity for the target system in the appropriate units from a look-up table, and calculating the number of partition processors. The number of partition processors equals the total number of system processors, times the partition workload capacity divided by the system work capacity. The calculated number of partition processors is tested to see if it is within a predetermined percentage of the next full processor increment. If within the predetermined percentage, then using dedicated processors is recommended, otherwise using shared processors is recommended. The calculated number of partition processors and the recommended use of shared or dedicated processors is displayed to the user for validation or changing of the values. After validation, the processors are configured according to the settings determined by the routine.

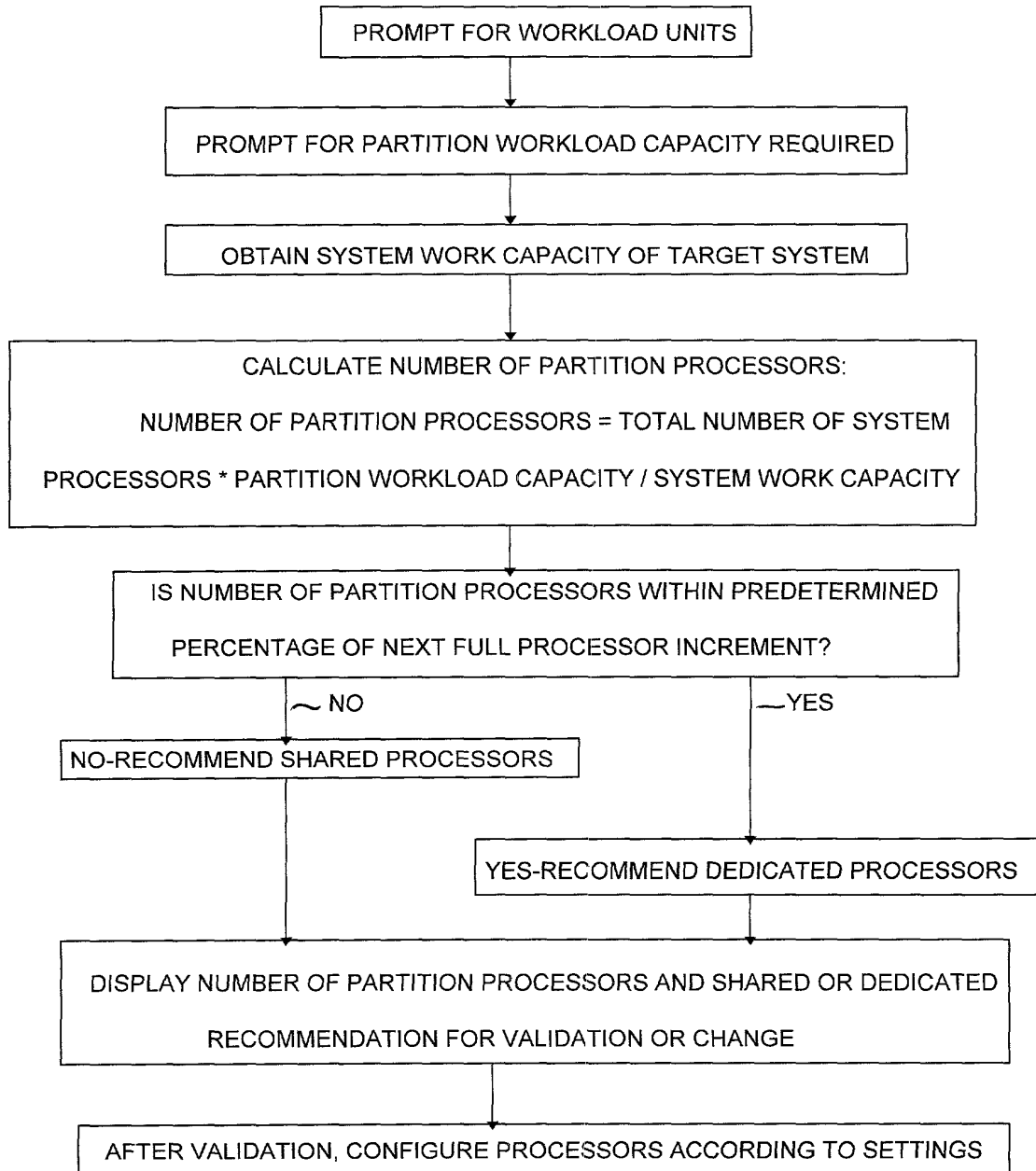


FIG. 1

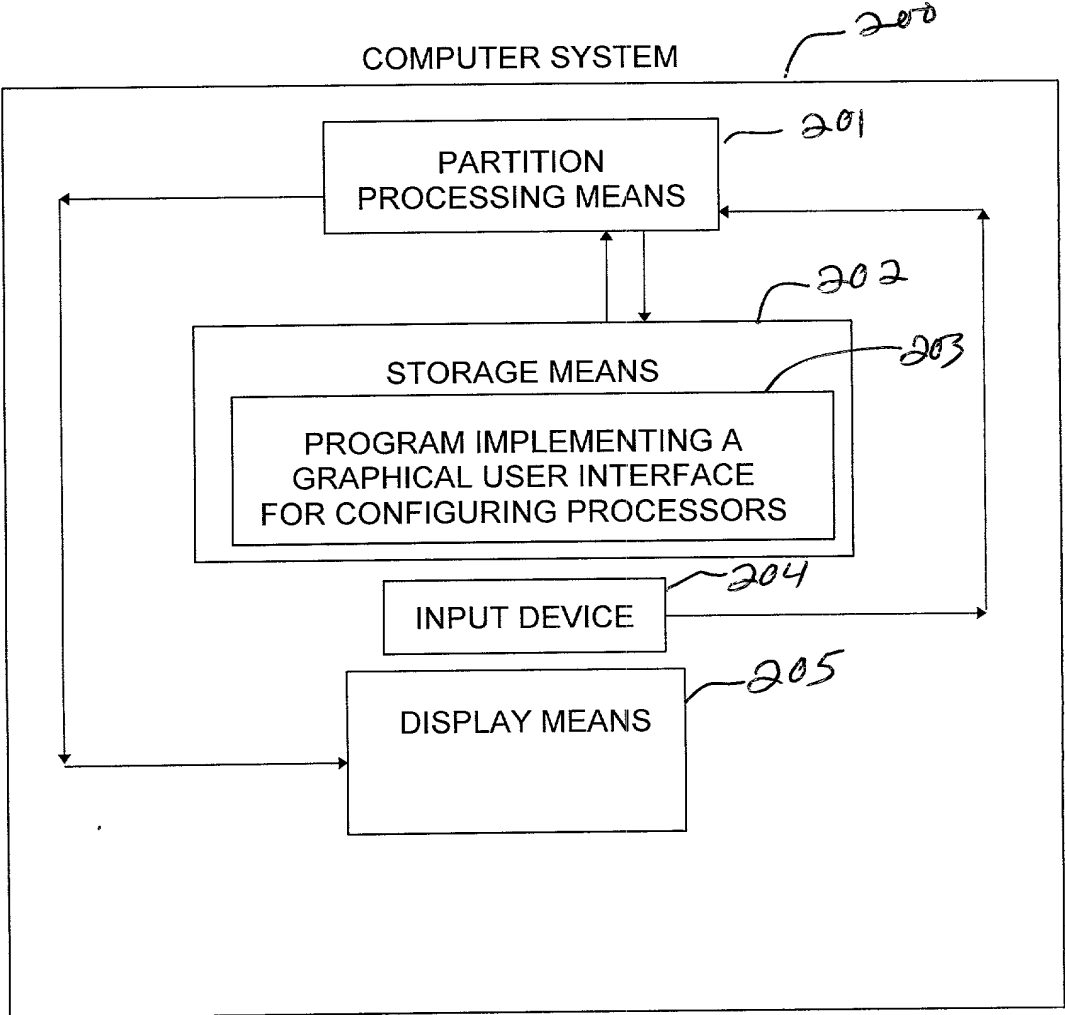


FIG. 2

DECLARATION AND POWER OF ATTORNEY FOR PATENT APPLICATION

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name. I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled:

TECHNIQUE FOR CONFIGURING PROCESSORS IN SYSTEM WITH LOGICAL PARTITIONS

the specification of which (check one)

X is attached hereto.

_____ was filed on _____ as
Application Serial No. _____
and was amended on _____
(if applicable)

I hereby state that I have reviewed and understand the contents of the above identified specification, including the claims, as amended by any amendment referred to above.

I acknowledge the duty to disclose information which is material to the patentability of this application in accordance with Title 37, Code of Federal Regulations, §1.56.

I hereby claim foreign priority benefits under Title 35, United States Code, §119 of any foreign application(s) for patent or inventor's certificate listed below and have also identified below any foreign application for patent or inventor's certificate having a filing date before that of the application on which priority is claimed:

Prior Foreign Application(s)			Priority Claimed	
			Yes	No
_____	_____	_____	_____	_____
(Number)	(Country)	(Day/Month/Year Filed)		

I hereby claim the benefit under Title 35, United States Code, §120 of any United States Application(s) listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code, §112, I acknowledge the duty to disclose information material to the patentability of this application as defined in Title 37, Code of Federal Regulations, §1.56(a) which occurred between the filing date of the prior application and the national or PCT international filing date of this application:

_____	_____	_____
(Application Serial No.)	(Filing Date)	(Status: Patented, Pending, Abandoned)

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

006260 "E4027960

POWER OF ATTORNEY: As a named inventor I hereby appoint the following attorney(s) and/or agent(s) to prosecute this application and transact all business in the Patent and Trademark Office connected therewith.

(List name and registration number)

William J. McGinnis, Jr. - 25,698
James R. Nock - 42,937
Steven W. Roth - 34,712

John E. Hoel - 26,279
Christopher A. Hughes - 26,914
Edward A. Pennington - 32,588
Joseph C. Redmond, Jr. - 18,753

Christopher H. Lynt - 33,619

Send Correspondence to:

Christopher H. Lynt
7502 Toll Ct.
Alexandria, VA 22306

Direct telephone Calls to: (703) 768-0956 [FAX: 703-721-3327]

Full name of sole or first Inventor: JEFFREY JAY SCHEEL

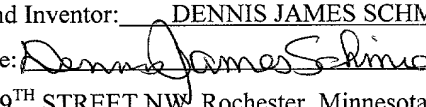
Inventor's signature:  Date: 9/28/2000

Residence: 6002 24TH AVENUE NW, Rochester, Minnesota (Olmsted County) 55901

Citizenship: U.S.A.

Post Office Address: (SAME AS RESIDENCE ADDRESS)

Full name of second Inventor: DENNIS JAMES SCHMIDT

Inventor's signature:  Date: 9/28/2000

Residence: 1880 49TH STREET NW, Rochester, Minnesota (Olmsted County) 55901

Citizenship: USA

Post Office Address: (SAME AS RESIDENCE ADDRESS)

☒ This declaration ends with this page.

006660" 64024960